

**Universitat de Lleida**

Escola Politècnica Superior

Enginyeria Tècnica en Informàtica de Sistemes

Treball de final de carrera

# Entorn web per a l'administració d'un cluster de computació distribuïda

Autor: Albert Ros Simon

Director: Fernando Guirado Fernández

Setembre del 2007

# Índex

<b>1</b>	<b>Introducció</b>	<b>5</b>
1.1	Administració i gestió d'un cluster de computadors . . . . .	5
1.2	Tipus de tasques realitzades en un cluster . . . . .	5
1.3	Tipus d'usuaris d'un cluster . . . . .	8
1.4	Formes de realitzar les tasques en un cluster . . . . .	8
1.5	Objectius . . . . .	10
1.5.1	Tipus de tasques . . . . .	11
1.5.2	Tipus d'usuaris . . . . .	13
<b>2</b>	<b>Implementació</b>	<b>17</b>
2.1	Escenari . . . . .	17
2.2	Requeriments . . . . .	18
2.2.1	Senzillesa . . . . .	19
2.2.2	Dinamisme . . . . .	20
2.2.3	Modularitat . . . . .	20
2.3	Entorn escollit . . . . .	21
2.4	Components necessaris . . . . .	21
2.4.1	LAMP . . . . .	22
2.4.2	Ganglia . . . . .	24
2.5	Instal·lació i primers passos . . . . .	29
2.5.1	Instal·lació LAMP . . . . .	30
2.5.2	Instal·lació Ganglia . . . . .	31
2.5.3	Instal·lació Drupal . . . . .	31
2.5.4	Aspectes de seguretat . . . . .	32
<b>3</b>	<b>Desenvolupament</b>	<b>34</b>
3.1	Mòduls . . . . .	34
3.1.1	Mòduls modificats . . . . .	34
3.1.2	Mòduls creats . . . . .	37
3.1.3	Tasques d'administració . . . . .	40
3.2	Gestió de la seguretat . . . . .	41
3.2.1	Contrasenyes d'usuaris . . . . .	41
3.2.2	Dades de formularis . . . . .	43
3.2.3	Cookies . . . . .	45
3.3	Integració amb Ganglia . . . . .	46

<i>ÍNDEX</i>	2
<b>4 Conclusions i treball futur</b>	<b>49</b>
<b>A Scripts d'instal·lació</b>	<b>52</b>
A.1 install_gmetad . . . . .	52
A.2 install_gmond . . . . .	53
A.3 install_drupal . . . . .	54
<b>B Ampliacions mitjançant mòduls</b>	<b>56</b>

## Índex de figures

1	Esquema dels elements de l'entorn . . . . .	11
2	Escenari de proves . . . . .	18
3	Tipus de mòduls d'un CMS . . . . .	20
4	Esquema de l'arquitectura/relacions de la plataforma LAMP . . .	24
5	<i>Ganglia</i> en mode Cluster . . . . .	25
6	<i>Ganglia</i> en mode Hosts . . . . .	26
7	<i>Ganglia</i> en mode Node . . . . .	27
8	<i>Ganglia</i> en mode Host . . . . .	28
9	Esquema de l'arquitectura de <i>Ganglia</i> . . . . .	29
10	Administració dels blocs . . . . .	46
11	Configuració dels blocs . . . . .	47
12	Resultat dels blocs . . . . .	48

**Índex de taules**

1	Usuaris predefinits a l'entorn . . . . .	13
2	Característiques dels clusters . . . . .	17
3	Tasques d'administració . . . . .	41

## 1 Introducció

En aquest capítol es fa una introducció sobre què consisteix l'administració d'un cluster de computadors. Es comenta quin tipus de tasques es realitzen en un cluster, de quina forma es poden executar aquestes tasques, i els tipus d'usuaris que s'hi poden trobar. Finalment es donen a conèixer els objectius d'aquest Treball de Final de Carrera.

### 1.1 Administració i gestió d'un cluster de computadors

Un cluster és un grup de computadors independents construïts amb components hardware comuns i en la majoria dels casos, software de lliure distribució. Aquests computadors s'interconnecten mitjançant xarxes d'alta velocitat.

Aquests grups de computadors s'utilitzen per a executar una sèrie d'aplicacions de forma conjunta, i es mostren com un únic sistema. En cada cluster hi ha un node que s'utilitza per a la seva administració, el servidor del cluster, també anomenat frontend quan ofereix una interfaz web.

La utilitat dels clusters és realitzar tasques que necessiten una capacitat de còmput que no es pot trobar en ordinadors comuns. Algunes d'aquestes tasques són càlculs aplicats a meteorologia, astronomia, química, electrònica o criptografia. L'administrador d'un cluster ha de tenir en compte alguns aspectes, una vegada s'hagi completat la instal·lació dels recursos bàsics de hardware i software. Aquests aspectes inclouen la configuració i administració de recursos mitjançant eines de software, la monitorització de les activitats i el registre dels events generats per l'execució de càlculs computacionals.

### 1.2 Tipus de tasques realitzades en un cluster

L'administració de clusters implica prendre mesures preventives i planificar tasques, que inclouen els següents aspectes:

- **Registre d'esdeveniments:** El maneig de logs, o el registre d'esdeveniments generats tant pel kernel del sistema operatiu com pels diferents serveis que estableixen comunicació entre els nodes, es pot realitzar mitjançant comandes del sistema operatiu per visualitzar els arxius de logs, o utilitzar eines específiques de monitorització. Gràcies a aquests registres es poden detectar els motius d'alertes o fallides que es produeixen en el sistema operatiu.

- Monitorització de l'estat dels recursos del cluster: La monitorització permet conèixer si tots els components hardware y software estan disponibles i operant de la forma esperada. S'ha d'assegurar que tots els components hardware estiguin disponibles durant l'arrancada del sistema operatiu (CPU's, memòria, discs, dispositius de xarxa, etc.), i que tots els serveis software, (planificadors de tasques, administradors de recursos, daemons de monitorització, etc.) s'executin correctament al cluster. A més, degut al mal funcionament d'alguns nodes, permet detectar colls d'ampolla en la xarxa, ja que entre els nodes no existiria bona comunicació.
- Recuperació davant fallides de hardware o software: L'administració del cluster implica resoldre problemes provocats per fallides de hardware i/o software, a vegades identificats gràcies al registre d'esdeveniments. Les fallides provocades pel hardware poden provocar que el cluster quedi inutilitzat. La recuperació davant fallides a nivell de hardware implica:
  1. Aïllar els components que hagin fallat per assegurar que no causen més impacte a les activitats del cluster.
  2. Manejar els components hardware de suport (backup), per poder fer recuperacions i minimitzar els efectes de la fallida.

Les fallides de components de software algunes vegades no tenen solució o forma de recuperació. La majoria d'aplicacions necessiten actualitzacions o noves versions per millorar o recuperar-se de bugs. Per això, si un component software falla s'han de buscar actualitzacions o informar a l'equip de desenvolupament de l'aplicació per a que en treguin alguna. Tot i això, també existeix alguna eina que intenta que les fallides en una màquina determinada no afectin al funcionament del cluster, intentant aixecant els serveis caiguts en una altra màquina. Aquest és el cas de l'eina *Fake* [1].

- Accounting: L'entorn Linux ofereix algunes alternatives per mantenir còpies d'un conjunt d'arxiu en diferents equips. La forma més fàcil i comú d'administrar aquestes còpies involucra l'ús d'una xarxa basada en servei per l'administració de comptes o registres d'usuari. Quan s'utilitza aquesta alternativa, cada ordinador realitza consultes a un servei central, el que maneja l'autorització, l'autenticació i la informació dels usuaris dins del sistema.

Per la configuració manual de clusters, els serveis més utilitzats són *NIS* (Network Information Service) i *LDAP* (Lightweight Directory Access Protocol). No obstant, també s'habiliten de forma automàtica amb els toolkits *OSCAR* [2] i *Rocks* [3] .

- Planificació de tasques i balanceig de càrrega: Les activitats de planificació i balanceig de càrrega que son crítiques per un entorn cluster són:
  - Determinar la disponibilidad dels nodes.
  - Configurar atributs dels nodes que siguin importants pel balanceig de càrrega.
  - Administrar usuaris i grups mitjançant quotes de disc.
  - Configurar i dissenyar polítiques.
  - Administrar reserves i recursos dedicats.
  - Monitoritzar i generar un historial d'utilització de recursos per usuaris i grups.
  - Administració del registre d'usuaris i grups d'usuaris.

Els sistemes més importants per a instal·lació automàtica de clusters, inclouen eines de monitorització, administració i registre d'events mitjançant paquets de distribució per sistemes *Windows* i *Linux*. Entre aquests sistemes es troben *OSCAR* i *Rocks*, que permeten l'ús d'eines de software que tenen propòsits específics com:

- Definició i administració de nodes.
- Administració de cues per lots (Batch Queue Management).
- Administració de recursos: grups NIS (Network Information Service), quotes de disc i CPU.
- Administració de les cues d'execució.
- Administració de serveis de resolució de noms : DNS (Domain Name System per clusters).
- Monitorització de càrrega.
- Registre d'usuaris.



### 1.3 Tipus d'usuaris d'un cluster

En qualsevol cluster es poden diferenciar 3 tipus d'usuaris, cadascun d'ells amb uns propòsits diferents. Aquests tipus d'usuaris són:

- Administrador: Aquest tipus d'usuari té accés a qualsevol de les tasques que es poden realitzar en un cluster, descrites a l'apartat 1.2 .
- Operaris: Els operaris s'encarreguen d'administrar només alguns aspectes del cluster, com poden ser l'administració d'usuaris o la gestió de recursos. D'aquesta forma, es poden assignar operaris especialitzats, i totes les tasques d'administració no recauen en l'usuari administrador.
- Usuaris: Els usuaris són els que utilitzen el cluster per a llençar processos i aprofitar-se dels seus recursos. Per a això hauran d'actuar segons unes polítiques d'accés que assegurin el seu treball sense interferències mútues.

### 1.4 Formes de realitzar les tasques en un cluster

Les formes de realitzar les tasques en un cluster es podrien diferenciar en 2 grups: segons el tipus i segons la forma. El primer grup es podria dividir entre eines generals i eines específiques. Les eines generals permeten realitzar diferents tipus de tasques, en canvi les específiques estan orientades a realitzar alguna tasca determinada, oferint més opcions que una eina general. El segon grup es divideix entre les eines basades en línia de comandes i les que incorporen un entorn gràfic. A continuació es llisten algunes eines específiques que permeten realitzar algunes tasques comentades en l'apartat 1.2 :

#### Registre d'esdeveniments

- LogCheck [4]
- LogSentry [5]
- LogDog [6]

En sistemes *GNU/Linux*, la comanda més utilitzada per visualitzar logs és *tail*.

**Monitorització i estat del cluster**

- MON [7]
- Big Brother [8]
- CluMon [9]
- Ganglia [29]
- Nagios [10]
- Supermon [11]
- Ldirectord [12]

**Recuperació davant fallides**

- Fake [1]

**Planificació de tasques i balanceig de càrrega**

- Ultra Monkey [13]

Un bon exemple del segon grup d'eines és el paquet *C3 - Cluster Command & Control*, de distribució lliure, que conté un conjunt d'utilitats basades en línia de comandes. Aquestes utilitats s'utilitzen per executar tasques comuns d'administració. Aquestes comandes es van dissenyar per oferir un entorn similar a les comandes que s'utilitzen en l'administració d'un sistema *UNIX* o *Linux*. Aquest paquet s'instal·la automàticament amb la distribució *OSCAR*, i incorpora les següents comandes:

- *cexec*: Executa una comanda en tots els nodes.
- *cget*: Copia arxius des d'una determinada ubicació dels nodes.
- *cpush*: Copia arxius cap a una determinada ubicació dels nodes.
- *ckill*: Permet finalitzar processos d'execució en els nodes.
- *crm*: Permet eliminar arxius i directoris dels nodes.
- *cshutdown*: Permet apagar, reiniciar o suspendre un node.
- *clist*, *cget*, *cnum*: Són utilitzats per fer consultes d'arxius de configuració.

Per acabar, un exemple d'eina específica basada en línia de comandes. El *PBS* (Portable Batch System) [15] és una eina per a l'administració de cues de treball, que permet enviar treballs a una cua per executar en una plataforma paral·lela, de forma molt simple. *PBS*, conté moltes comandes, i les que s'utilitzen amb més freqüència són:

- *qsub*: Envia una tasca al sistema de cues.
- *qdel*: Elimina una tasca de la cua.
- *qstat*: Mostra l'estat de les tasques de la cua.
- *xpbs*: Una interfície gràfica pels usuaris de PBS.

## 1.5 Objectius

L'objectiu d'aquest Treball de Final de Carrera és el de crear un entorn web que permeti administrar i monitoritzar un cluster de computadors. Això facilitarà la feina a l'administrador, ja que el permetrà accedir al cluster per executar tasques d'administració i, a més, podrà visualitzar els recursos del cluster de forma gràfica des del navegador web. Gràcies a l'entorn web es podrà administrar el cluster des de qualsevol màquina de la xarxa, sense necessitat d'identificar-se com a administrador al cluster.

A la Figura 1 es mostra un esquema de la forma en que interactuen els diferents elements que formen part de l'entorn. El quadre puntejat senyalitza els elements existents abans de crear l'entorn web.

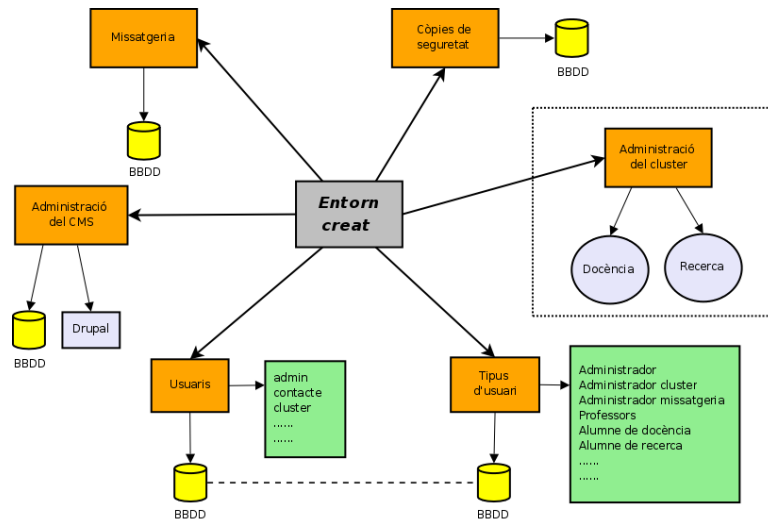


Figura 1: Esquema dels elements de l'entorn

En els següents punts es presenten les principals característiques del disseny de l'entorn.

### 1.5.1 Tipus de tasques

A continuació es mostren les tasques a realitzar referents a l'entorn del cluster i a l'entorn web. Les primeres fan referència a les tasques realitzades habitualment en un cluster, mentre que les segones són les que apareixen en el moment que s'implementa un entorn web.

#### *Entorn cluster*

- Gestió de còpies de seguretat: Copiar carpetes especificant la carpeta origen, el destí (arxiu comprimit) i el moment que s'efectuarà la còpia.
- Gestió d'usuaris: Afegir i usuaris del sistema. Modificar el grup al que pertanyen els usuaris.
- Gestió de grups: Llistar, afegir i grups del sistema.
- Monitorització del cluster: Obtenir informació dels recursos del cluster en temps real a través de l'eina *Ganglia*.

- Comunicació entre usuaris: Enviament de missatges entre els usuaris i l'administrador del cluster.

### ***Entorn web***

- Control d'accés: Donar o treure permissos als usuaris per a accedir a determinades funcions de l'entorn.
- Configuració compte d'usuari: Modificar aspectes per a cada usuari, com l'idioma de l'entorn, l'adreça de correu electrònic o la contrasenya.
- Creació de contingut: Crear contingut que es visualitzarà a la pàgina principal de l'entorn. Es poden crear diferents tipus de contingut:
  - Història: Les històries són articles que tenen títol i cos. Es fan servir per crear notícies que veuran tots els usuaris que tinguin accés al contingut.
  - Esdeveniment: Els esdeveniments són històries amb una data d'inici i final. Quan són creats, apareixen al calendari d'esdeveniments i quan falten pocs dies per a la data d'inici, apareixen al bloc *Pròxims esdeveniments*.
  - Tasca: Les tasques serveixen per crear esdeveniments propis de l'usuari. Només es mostren a l'usuari que les hagi creat.
  - Tasca grup: Les tasques de grup serveixen per crear esdeveniments per grups d'usuaris. Només es mostren als usuaris que pertanyen al mateix grup (rol) que l'usuari que les crea.
- Gestió de contingut: Llistar, editar i el contingut del lloc web. Afegir nous tipus de contingut.
- Administració de mòduls: Habilitar o deshabilitar els mòduls per a modificar la funcionalitat del lloc.
- Administració de blocs: Configurar en quina regió i per a quins usuaris es mostren algunes funcionalitats del lloc.
- Administració de menús: Controlar els elements que es mostren en el menú de navegació.
- Administració de temes: Seleccionar el tema de visualització i configurar aspectes com el logotip o el nom del lloc.

- Gestió de registres: Supervisar el lloc web, capturant les incidències del sistema en un registre que podrà ser revisat per l'administrador.
- Configuració de l'entorn web: Configurar aspectes com la data, l'hora, els idiomes disponibles i el sistema de fitxers.

### 1.5.2 Tipus d'usuaris

A l'entorn de treball d'un cluster hi treballen diferents tipus d'usuaris, com s'ha vist a l'apartat 1.3, per aquest motiu cal poder definir-los i identificar-los. En aquest apartat només es tracta els diferents tipus d'usuaris des del punt de vista de l'entorn web.

Dins l'entorn web s'ha diferenciar entre els usuaris i els rols d'usuari. En el moment de donar d'alta un usuari, se li assigna un o més rols. Aquests rols són els que determinen a quines funcions pot tenir accés cada usuari. En qualsevol moment es poden modificar tant els permissos que té cada rol d'usuari com els rols que té assignats un usuari i, automàticament, s'actualitzen els permissos d'accés. A més, l'administrador sempre podrà afegir nous tipus d'usuari a l'entorn. A la Taula 1 es mostra una relació dels usuaris que es crearan de forma predefinida a l'entorn web.

Tipus d'usuari	Nom d'usuari	Rol d'usuari
Administrador	admin	admin
Administrador del cluster	cluster	clusteradmin
Administrador de missatgeria	contacte	contacteadmin

Taula 1: Usuaris predefinits a l'entorn

A continuació es detallen els tipus d'usuari que es defineixen, amb el rol que se'ls hi ha assignat, i la seva funcionalitat:

**Administrador (*Rol admin*):** És l'administrador principal de l'entorn web, i la seva funció és gestionar aspectes generals de l'entorn web. Té accés a les següents operacions:

- Fer còpies de seguretat (instantànies o programades)
- Llistar còpies de seguretat pendents
- Accedir a missatgeria

- Afegir i usuaris (entorn web)
- Afegir i rols d'usuari (entorn web)
- Afegir i usuaris (sistema Linux)
- Afegir i grups (sistema Linux)
- Control d'accés segons rols d'usuari
- Accedir al contingut: història, esdeveniment, tasca i tasca de grup
- Crear contingut: història, esdeveniment, tasca i tasca de grup
- Gestionar el contingut
- Administrar mòduls
- Administrar blocs
- Administrar menús
- Administrar temes
- Administrar esdeveniments
- Administrar paràmetres d'usuari
- Gestionar els registres de l'entorn web
- Configurar l'entorn web: manteniment, sistema de fitxers, informes d'error, informació del lloc.....
- Configurar compte d'usuari

**Administrador del cluster (*Rol clusteradmin*):** És l'administrador de l'entorn del cluster i, a diferència de l'anterior, aquest principalment s'utilitza per a gestionar el cluster. Les seves principals funcions són:

- Fer còpies de seguretat (instantànies o programades)
- Llistar còpies de seguretat pendents
- Accedir a missatgeria
- Afegir i usuaris (entorn web)

- Afegir i rols d'usuari (entorn web)
- Afegir i usuaris (sistema Linux)
- Afegir i grups (sistema Linux)
- Control d'accés segons rols d'usuari
- Accedir al contingut: història, esdeveniment, tasca i tasca de grup
- Crear contingut: esdeveniment, tasca i tasca de grup
- Configurar compte d'usuari

**Administrador de missatgeria (*Rol contacteadmin*):** És l'usuari encarregat de gestionar els missatges que es reben de la resta d'usuaris que tenen habilitat el servei de missatgeria i el seu ús es limita a:

- Gestionar missatgeria (enviar i missatges)
- Accedir al contingut: història i esdeveniment
- Configurar compte d'usuari

**Professor (*Rol professors*):** Aquest tipus d'usuari té accés a les següents operacions:

- Accedir a missatgeria
- Accedir a monitorització del cluster de recerca
- Accedir a monitorització del cluster de docència
- Accedir al contingut: història, esdeveniment, tasca i tasca de grup
- Crear contingut: esdeveniment, tasca i tasca de grup
- Configurar compte d'usuari

**Alumne de Recerca (*Rol alumner*):** Alumne que utilitza el cluster de recerca, té permissos per a:

- Accedir a missatgeria
- Accedir a monitorització del cluster de recerca



- Accedir al contingut: història, esdeveniment i tasca
- Crear contingut: tasca
- Configurar compte d'usuari

**Alumne de Docència (*Rol alumned*):** Alumne que utilitza el cluster de docència. té permissos per a:

- Accedir a missatgeria
- Accedir a monitorització del cluster de docència
- Accedir al contingut: història, esdeveniment i tasca
- Crear contingut: tasca
- Configurar compte d'usuari

**Usuari no registrat:** És l'usuari que no està registrat a l'entorn o el que encara no ha introduït el seu nom d'usuari i contrasenya per a identificar-se.

- Accedir al contingut: història i esdeveniment

## 2 Implementació

En aquest tema s'expliquen els plantejaments que han sorgit durant la implementació del projecte i les solucions que s'han aplicat per tal de resoldre'ls. Inicialment es presentarà l'escenari de partida on actuarà l'entorn desenvolupat, a continuació es presentaran els requeriments necessaris per aquest entorn i els seus components i finalment s'abordarà la seva instal·lació.

### 2.1 Escenari

L'escenari d'aquest projecte es situa al laboratori del Grup de recerca de Computació Distribuïda (*GCD*) de la UdL. En aquest laboratori es treballa amb dos clusters anomenats AEC2 i Loras. El cluster AEC2 està destinat a la docència, on alguns alumnes (a més dels professors) hi tenen accés per a realitzar diferents tasques. En canvi, el Loras, o cluster recerca, està destinat a la investigació. Tot i així, també es fa necessari donar accés a alguns alumnes.

Cada cluster (frontend + nodes) forma una xarxa Gigabit, i aquestes 2 xarxes estan interconnectades a través d'un equip de xarxa. A la Taula 2 es mostren les característiques principals de cada cluster.

	Loras (Recerca)	AEC2 (Docència)
Sistema Operatiu	Fedora Core 5	Rocks
Nombre de nodes	18	5
IP	172.16.104.44	172.16.3.100

Taula 2: Característiques dels clusters

La màquina frontend del cluster Loras anomenada *mm* fa de servidor per a l'entorn que es desenvolupa en aquest projecte. Això vol dir que sobre ell correrà el servidor web, i totes les operacions d'administració només afectaran al cluster Loras, tal com es mostra en l'esquema de la Figura 2 .

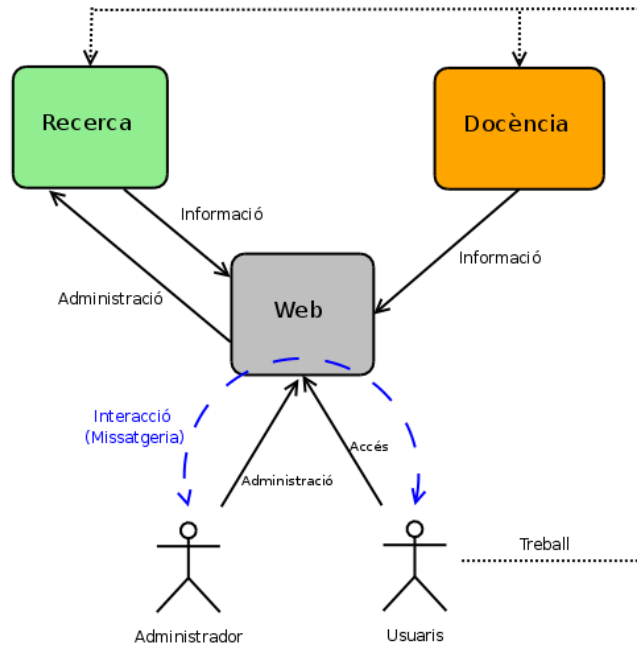


Figura 2: Escenari de proves

D'altra banda, la única funció que es realitzarà sobre l'AEC2 serà la de monitorització a través del *Ganglia Web Frontend*. Aquest cluster disposa d'una instal·lació de *Rocks*, distribució GNU/Linux específica per a clusters. Aquesta distribució incorpora una eina de monitorització de recursos anomenada *Ganglia*, que es tractarà en capítols posteriors.

## 2.2 Requeriments

El primer pas que s'ha de fer és escollir el conjunt d'eines amb les que implementar el sistema desitjat. Aquest és un pas important ja que s'ha d'arribar a un punt en què sigui viable i rentable portar el projecte endavant.

Per a dur a terme aquesta elecció s'ha realitzat un estudi de les possibilitats que hi ha actualment, per tal de trobar els llenguatges i les eines més adients. Aquests llenguatges i eines hauran de complir una sèrie de requeriments, els quals són detallats en els següents punts.

### 2.2.1 Senzillesa

Per a crear l'entorn web desitjat, existeixen dues possibilitats: començar desde zero o utilitzar algun Sistema de Gestió de Contingut (CMS). Començar desde zero implica haver de programar una gran quantitat de codi, amb la gran probabilitat que apareguin errors i bugs de funcionament. Reparar aquests errors pot fer perdre molt temps i això pot repercutir en que un projecte sigui considerat com a viable o no viable.

Per evitar això s'ha decidit optar per la segona opció, utilitzar un CMS ja existent. Al mercat hi ha una gran oferta de CMS [16, 17]. Aquestes eines permeten crear un entorn web d'una forma senzilla, i fan que no sigui necessari tenir nocions de cap llenguatge de programació. Al ser una eina enfocada a un tipus d'usuaris amb pocs coneixements informàtics, fa que el seu funcionament sigui molt senzill. Això és un element a tenir en compte, ja que tant l'administrador com els usuaris aprenen a moure's ràpidament per l'entorn.

Un CMS permet la creació i administració de continguts dins d'un entorn web. Tot el contingut s'emmagatzema en una o vàries bases de dades, i això afavoreix a que el sistema pugui tractar de manera independent el contingut i el disseny. Així, en qualsevol moment es pot canviar el disseny, mitjançant temes, sense haver de donar-li un nou format al contingut que existeixi. Existeixen diferents tipus de CMS, que es poden agrupar en les següents categories:

- Espais web: Espai web amb contingut i funcionalitat diversa.
- Fòrums: Lloc que permet la discussió en línia entre els seus usuaris, d'algun tema determinat.
- Blogs: Publicació de notícies o articles, amb espai per comentaris i discussions.
- Wikis: Espai web on tots els usuaris poden col·laborar en la creació dels articles.
- Galeria: Publicació i administració de contingut audiovisual.

Com es pot observar, existeixen diferents possibilitats, segons el tipus d'entorn que es desitgi crear. Quan es desitja un entorn obert, és a dir, que no només tingui un tipus de contingut, la millor opció és la primera, un espai web que permeti la possibilitat de configurar les funcions requerides. Dintre d'aquest tipus existeix un gran ventall de possibilitats, i

### 2.2.2 Dinamisme

Un sistema dinàmic és aquell que presenta algun canvi cada cert temps.

Per a un entorn web on la intenció és actualitzar el contingut contínuament, el dinamisme és un concepte obligat. Actualment la majoria d'entorns web disposen d'aquesta característica, encara que els usuaris o propis administradors no en siguin conscients. Uns clars exemples podrien ser els fòrums de debat o els blogs, on constantment les dades són actualitzades. Aquest tipus d'espais seria inviable d'implementar-los en un sistema estàtic. Un entorn web dinàmic està lligat a una o vàries bases de dades, on s'inseriran, actualitzaran les dades.

### 2.2.3 Modularitat

El gran avantatge de treballar en un sistema modular, és que sempre es podran afegir noves funcionalitats a l'entorn mitjançant mòduls. Això permet que una vegada instal·lat i en funcionament, el sistema podrà ser actualitzat de forma còmoda i sense tenir la obligació de modificar el codi que ja s'està utilitzant.

Aquests tipus de sistemes, normalment estan formats per mòduls bàsics i mòduls addicionals. Els mòduls bàsics són aquells mòduls necessaris per al funcionament bàsic del sistema, és a dir, que són el seu nucli. En canvi, els mòduls addicionals són aquells que desenvolupen els membres de les comunitats d'usuaris i després comparteixen amb la resta d'usuaris, passant per unes normes d'estandarització i seguretat.

A la Figura 3 es pot observar un exemple del tipus de mòduls que poden existir en un CMS.

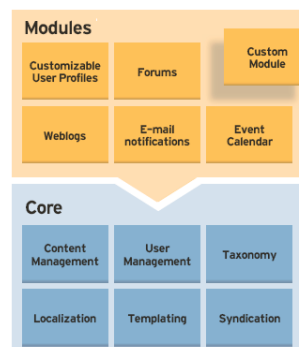


Figura 3: Tipus de mòduls d'un CMS

### 2.3 Entorn escollit

A l'hora d'escollir entre un dels CMS per a crear espais web s'han tingut en compte les següents possibilitats:

- PHP-Nuke [18]
- Postnuke [19]
- Mambo [20]
- Joomla! [21]
- Drupal [22]

Després d'haver-los estudiat, el CMS escollit ha sigut Drupal i els factors decisius de l'elecció són els següents:

- Ofereix gran dinamisme.
- Facilitat d'aprenentatge.
- Instal·lació àgil.
- Configuració senzilla.
- Gran quantitat de mòduls i temes. [23, 24]
- Gran comunitat d'usuaris. [25, 26, 27]
- Estabilitat.
- Actualització contínua.
- Bona documentació.
- És considerat un dels CMS que aporta més seguretat.

### 2.4 Components necessaris

En aquest capítol es comentaran quins són els components necessaris per a que totes les característiques de l'entorn funcionin correctament. Aquí es mencionaran aquells components que s'han hagut d'instal·lar apart del CMS Drupal.

### 2.4.1 LAMP

El sistema escollit per fer córrer l'aplicatiu web al servidor és el conegut com a LAMP [28]. LAMP és una plataforma de desenvolupament d'aplicacions web de codi obert, i el seu acrònim prové de Linux + Apache + MySQL + PHP (la lletra P també pot fer referència a Perl o Python)

**Linux:** És el sistema operatiu instal·lat al servidor, sobre el que funcionen les altres aplicacions de la plataforma LAMP. L'expressió "Linux" és utilitzada per referir-se a les distribucions GNU/Linux, col·leccions de software que contenen gran quantitats de paquets a més del nucli (Linux). El programari que acostumen a incloure consta de una gran varietat d'aplicacions, com entorns gràfics, paquets ofimàtics, servidors web, reproductors multimèdia, etc. Per a montar un sistema LAMP qualsevol distribució Linux és vàlida, sempre i quant tots els altres elements de l'arquitectura LAMP hi siguin presents.

**Apache:** Apache és el servidor web de la plataforma LAMP, i tot i ser software lliure, pot funcionar també sobre sistemes operatius propietaris com Windows o diferents versions propietàries d'UNIX. Potser aquest és el motiu, unit a la seva robustesa i ventall de funcionalitats, que sigui el servidor web amb més quota de mercat.

**MySQL:** Sistema de gestió de bases de dades basat en SQL (Structured Query Language). És un sistema multifil i multiusuari, i la seva implementació client-servidor consta d'un servidor i diferents clients. Aquest gestor és considerat com un dels més ràpids i robustos, tant per a petits com per a grans volums de dades. Les principals característiques són:

- Velocitat i robustesa.
- Pot treballar en diferents plataformes i sistemes operatius diversos.
- Sistema de contrasenyes i privilegis molt segur i flexible.
- Encriptació de contrasenyes a través de la xarxa.
- Registres de longitud fixa o variable.
- Els clients utilitzen sockets TCP o UNIX Socket per connectar-se al servidor.

**PHP:** És un llenguatge interpretat d'alt nivell que s'executa en el servidor, i pot ser implementat en pàgines HTML o en forma d'script. Està orientat al desenvolupament d'aplicacions per servidors, o creació de contingut dinàmic per llocs web. Les característiques principals d'aquest llenguatge són:

- Llenguatge senzill i de sintaxi còmoda, similar a C, C++ i Java.
- Llenguatge de programació orientat a objectes.
- Tot i ser interpretat, resulta molt ràpid quan s'integra amb Apache.
- Accés a gran quantitat de gestors de bases de dades: MySQL, ODBC, Oracle, DB2, dbm, adabas, Informix, Microsoft SQL Server, dBase, MS Access, etc.
- Disposa de suport IMAP, SMTP, FTP, SNMP, XML, NNTP, PDF, generació dinàmica de gràfics, etc.

Alguns dels avantatges del sistema LAMP són:

- En tot moment s'està fent referència a Programari Lliure, el que implica una gran reducció de costos, al no haver de pagar cap tipus de llicència per instal·lar aquest programari. Això és un factor improtant que facilita que el projecte es pugui dur a terme.
- El manteniment d' un sistema LAMP es redueix a actualitzar paquets que es poden descarregar lliurement d' Internet.
- Amb una bona administració, s'arriba a un nivell de seguretat molt elevat, ja que en cas declarar-se alguna alerta, s'alliberen actualitzacions molt ràpidament.
- Aquests 4 productes poden funcionar en una àmplia gama de maquinari, donant garanties d'estabilitat tant en equips amb amplis recursos com en equips més limitats.
- Es crea un producte final molt flexible, ja que es soporta una gran quantitat de tecnologies addicionals (PDF, FTP, Flash, etc.).
- El servidor pot incloure un tallafocs per evitar intrusions i assegurar la continuïtat d'operació i, a més, és immune a més del 99% dels virus a Internet, ja que aquests van destinats a atacar altres sistemes operatius més estesos.



Un dels objectius és crear una web amb contingut dinàmic, amb contínues modificacions de les dades. La combinació del llenguatge PHP amb una base de dades MySQL permet fer això de forma transparent per a l'usuari.

A la Figura 4 es mostra la relació entre els diferents elements de l'arquitectura LAMP.

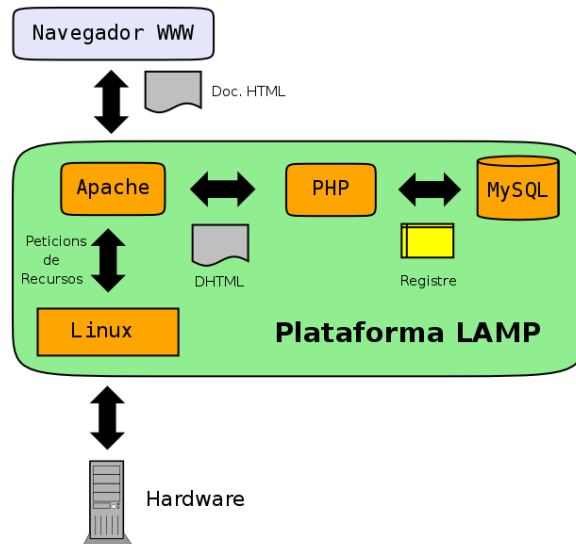


Figura 4: Esquema de l'arquitectura/relacions de la plataforma LAMP

Quan l'usuari fa una petició al servidor per a que li envii una pàgina web, al costat del servidor s'executa un script, capaç de connectar amb una base de dades per a accedir a la informació necessària. Una vegada s'accedeix a la informació, el servidor Apache mostra la informació en un format que el navegador web de l'usuari pot interpretar.

A més de tot això, el servidor haurà de tenir instal·lat el dimoni *sshd* per a fer possibles algunes operacions, tal com s'explica en el capítol 3.1.3.

## 2.4.2 Ganglia

*Ganglia* [29] és una eina de monitorització per a clusters que permet rebre informació en temps real de l'estat de cadascun dels seus nodes. A més de tenir

controlat en tot moment si cada node està operatiu o no, permet obtenir informació sobre aspectes que van des del temps que el node porta encès fins la quantitat de recursos que estan sent utilitzats. A la Figura 5 es mostra una captura de *Ganglia* en un mode de vista que contempla el cluster com una sola màquina. En ella es mostren gràfiques de càrrega de processos, CPU, memòria i trànsit de xarxa.

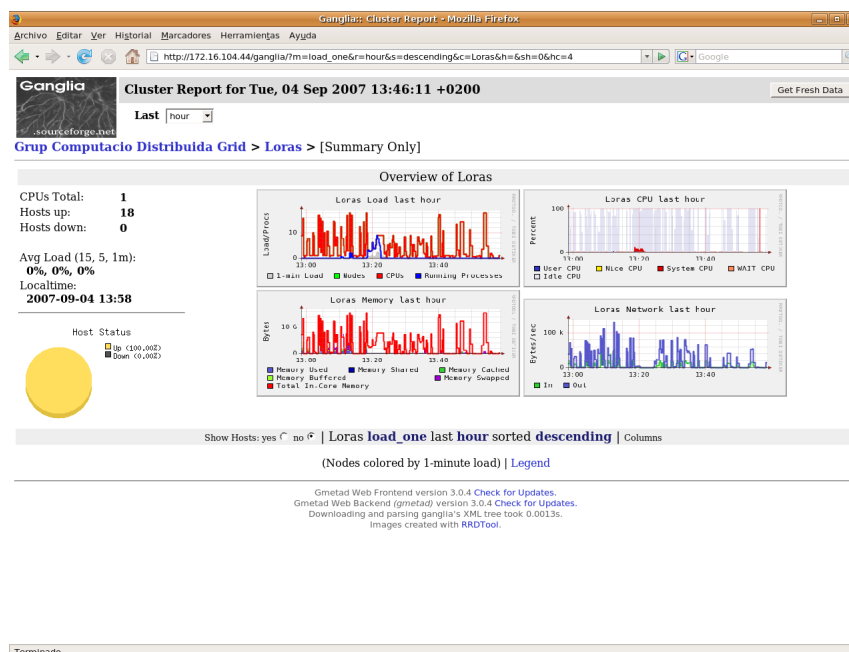


Figura 5: *Ganglia* en mode Cluster

A la Figura 6 es mostra la mateixa informació, pero aquesta vegada en mode de vista de hosts. Això vol dir que mostra la informació de cada host individualment.

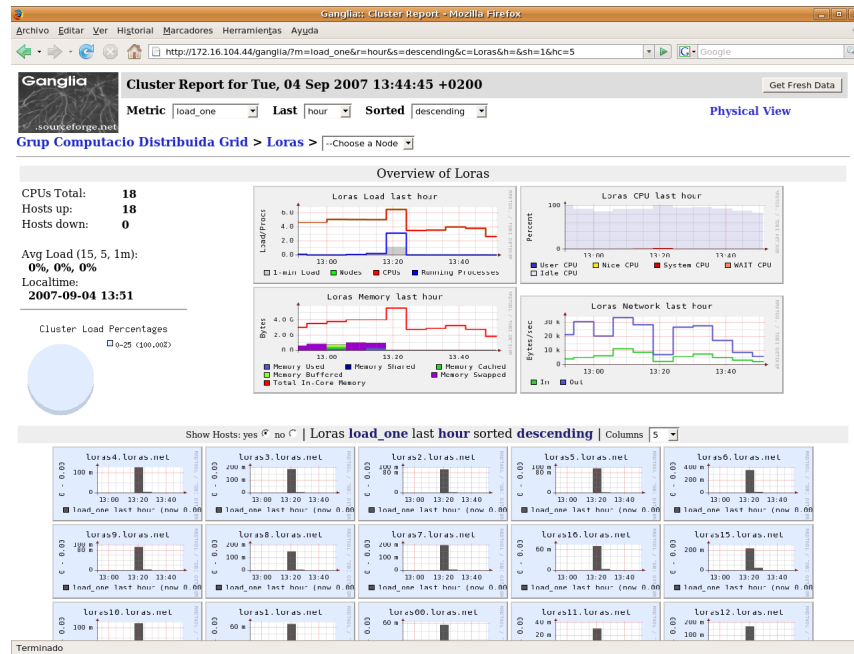


Figura 6: Ganglia en mode Hosts

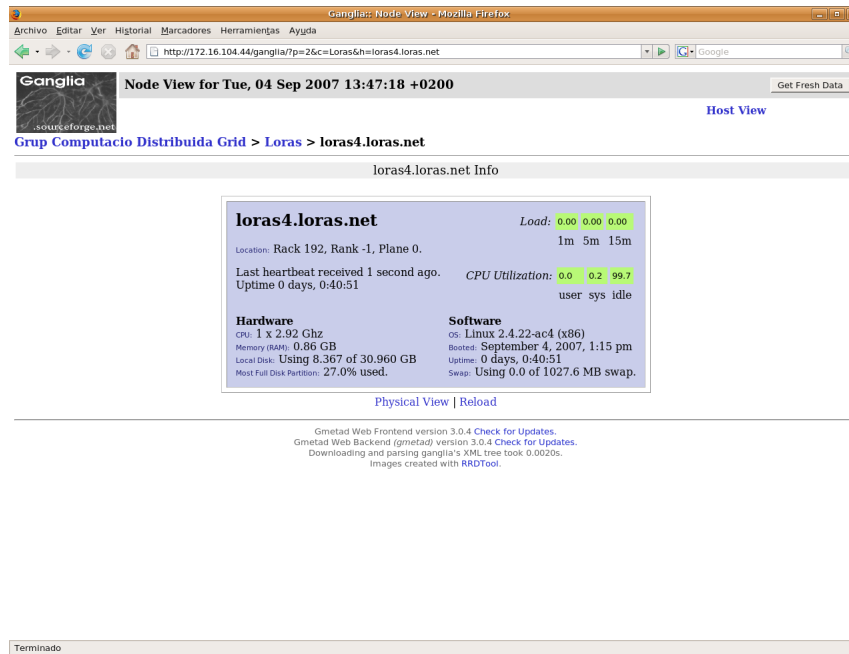
També és possible accedir a la informació d'un node determinat, fent click sobre la seva gràfica corresponent.

Quan s'accedeix a la pàgina d'un node, s'obté informació tant del seu hardware (nombre CPU's, velocitat CPU's, memòria) com del seu software (sistema operatiu, versió, dimonis, temps en actiu). A la Figura 7 es mostra aquesta informació en mode de vista Node.



Figura 7: Ganglia en mode Node

Si s'opta per l'opció de mode de vista Host, a més es mostrarà informació de la localització d'aquesta màquina i de la capacitat i espai ocupat de disc dur, tal i com es pot observar en la Figura 8 .

Figura 8: *Ganglia* en mode Host

El funcionament de Ganglia es basa en els seus dos dimonis, el *gmetad* i el *gmond*:

- Ganglia Meta Daemon (*gmetad*): Aquest dimoni corre a la màquina que fa el servei de frontend, és a dir, recullir dades dels nodes del cluster. La informació s'obté via XML en intervals regulars des dels nodes, i s'emmagatzemen en una base de dades Round-Robin per a ser compartida amb el servidor web o amb altres frontends on corri el dimoni *gmetad*. Això permet que des d'un frontend es puguin monitoritzar diferents clusters a la vegada.
- Ganglia Monitoring Daemon (*gmond*): És el dimoni que haurà de córrer en cadascun dels nodes que es vulguin monitoritzar.

A la Figura 9 es mostra la relació entre els dimonis de *Ganglia*. En ella s'aprecia com el dimoni *gmetad* de cada cluster obté la informació dels seus nodes a través del *gmond*. *Ganglia* és capaç de treballar en un mode d'operació de backup anomenat failover, en qual les funcions d'un component del sistema són

assumides per un segon component quan el primer no es troba disponible. Així, en un cas de fallida de connexió entre *gmetad* i *gmond*(1) d'un node determinat, *gmond*(1) envia la informació a un altre *gmond*(2), per a que *gmetad* la pugui llegir des d'aquest últim. La informació entre dimonis *gmond* s'envia en llenguatge XDR a través d'UDP.

L'arquitectura de *Ganglia* permet el funcionament de diferents clusters en cascada, de manera que el client pugui connectar-se a un servidor principal per a monitoritzar diferents clusters. Les dades que el dimoni *gmetad* obté d'altres deimonis *gmetad* o *gmond* són estructurades en llenguatge XML i transportades mitjançant TCP.

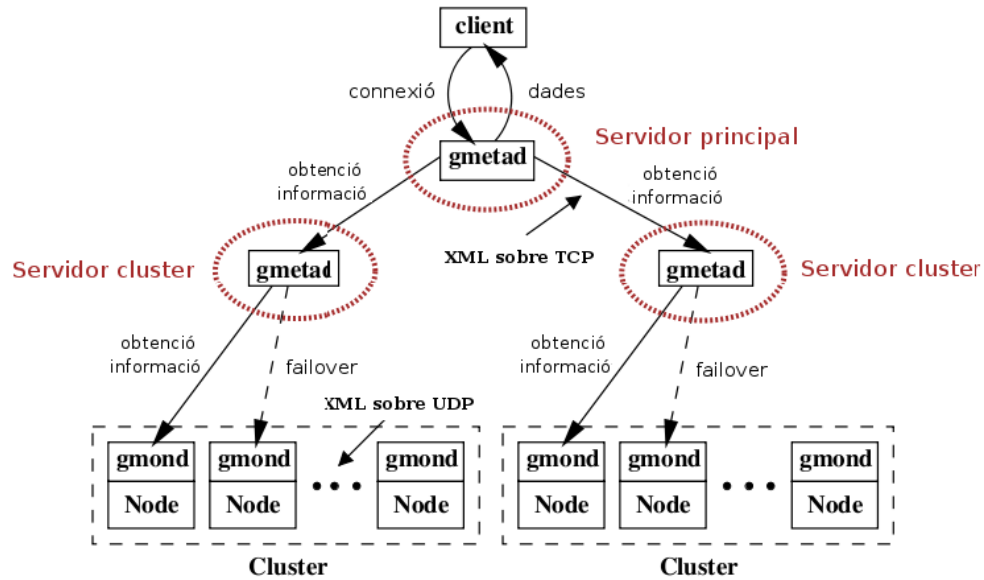


Figura 9: Esquema de l'arquitectura de *Ganglia*

## 2.5 Instal·lació i primers passos

En aquest apartat es detallen els passos a seguir per a instal·lar el sistema correctament. Els scripts que es mencionen en els capítols 2.5.2 i 2.5.3 es poden trobar en l'Annex A .

### 2.5.1 Instal·lació LAMP

En un sistema LAMP es pot diferenciar clarament entre el sistema operatiu del servidor (GNU/Linux) i la resta dels components (Apache, MySQL i PHP). Normalment, la instal·lació del sistema LAMP es centra en aquests últims components, ja que es realitza en un ordinador que ja té instal·lat GNU/Linux. Els gestors de paquets més comuns inclosos en les distribucions GNU/Linux són yum i apt-get. Aquests gestors faciliten la feina a l'hora de buscar, instal·lar i qualsevol tipus de programari a través d'internet. Per a instal·lar els paquets necessaris amb aquests gestors, només s'han d'executar les següents comandes:

```
sudo yum install httpd php php-gd php-mysql mysql-server  
sudo apt-get install apache2 php5 libapache2-mod-php5  
php5-common php5-mysql mysql-server
```

Una vegada instal·lats aquests paquets, s'hauran de llençar els dimonis que faràn que els serveis estiguin disponibles:

```
service httpd start  
service mysqld start
```

o bé:

```
/etc/init.d/httpd start  
/etc/init.d/mysqld start
```

A més, es poden configurar aquests serveis per a que corrin automàticament cada vegada que el servidor s'encengui, en diferents modes d'arrancada (runlevel's), per exemple, amb interfície gràfica o sense:

```
chkconfig --level 345 httpd on  
chkconfig --level 345 mysqld on
```

Per a comprovar si el servidor LAMP s'ha instal·lat i està corrent correctament, només caldrà adreçar-se amb el navegador a <http://localhost>, i si apareix la pàgina d'informació del servidor Apache, el sistema ja estarà llest.

### 2.5.2 Instal·lació Ganglia

Per a la instal·lació de *Ganglia* al frontend cal:

- Descarregar l'arxiu d'instal·lació des de la web del projecte Ganglia [29].
- Compilar l'arxiu d'instal·lació al sistema.
- Instal·lar Ganglia amb l'opció *gmetad*.
- Modificar l'arxiu de configuració del *gmetad*, que es troba a */etc/gmetad.conf*.
- Instal·lar el *Ganglia Web Frontend*, per a poder mostrar la informació rebuda en format de web dinàmica.
- Instal·lar l'eina RRDtool, utilitzada per a crear imatges a partir de la informació emmagatzemada a les bases de dades Round Robin.

Per a això s'ha creat l'script *install\_gmetad*, que es troba a l'Annex A.1. Aquest script, a més del dimoni *gmetad*, instal·larà els elements necessaris per a poder aprofitar la potència d'aquesta eina per a l'entorn web.

Per a instal·lar el dimoni *gmond* als nodes del cluster, des de cada node s'haurà d'executar l'script creat per a tal fi, l' *install\_gmond*. Aquest script es troba a l'Annex A.2 .

### 2.5.3 Instal·lació Drupal

Per a instal·lar Drupal s'ha de descarregar l'arxiu d'instal·lació des de la secció de descàrregues de la web oficial de Drupal [30]. Una vegada descarregat l'arxiu, s'ha de descomprimir a la carpeta on s'ubiquen els arxius del servidor web i després ja es pot accedir a Drupal a través del navegador web. Al accedir-hi per primera vegada, es demana a l'usuari que modifiqui els permisos d'algun arxiu de configuració i que afegeixi algun directori dins l'estructura de fitxers de Drupal, a més de seleccionar la base de dades que s'utilitzarà.

Per agilitzar aquest procés s'ha creat l'script d'instal·lació *install\_drupal*, inclòs en l'Annex A.3, que permetrà instal·lar l'entorn web al servidor en qualsevol moment i amb més comoditat.

L'script ha estat implementat per executar-se en la distribució GNU/Linux Fedora. En cas d'instal·lar-se en una altra distribució, caldria modificar alguns paràmetres. Per exemple, si la instal·lació es fa en una distribució basada en Debian, el gestor de paquets serà *apt-get* en lloc de *yum*, i les variables d'entorn seran:



```
usuari="www-data"  
home="var/www/"  
shell="/bin/bash"
```

La primera part de l'script s'encarrega de descomprimir l'estructura de directoris i arxius necessària per al funcionament del gestor de continguts a la carpeta del servidor web, i d'afegir la base de dades necessària al servidor MySQL. Aquesta base de dades està emmagatzemada en un arxiu, i ha sigut exportada des del servidor original des d'on s'ha implementat el sistema.

La segona part de l'script és necessària per a realitzar tasques pròpies de l'administració del sistema. Així es permetrà l'accés al sistema des de l'entorn web mitjançant el protocol ssh, tal com s'explica més detalladament en el següent capítol. Els passos que realitza l'script són els següents:

1. L'usuari apache afegeix la màquina localhost a l'arxiu *known\_hosts* ubicat a la carpeta *ssh* del seu home.
2. L'usuari apache crea una clau pública i una clau privada per a les comunicacions.
3. L'usuari root afegeix la clau privada de l'usuari apache a l'arxiu *authorized\_keys* de la carpeta *ssh*, ubicada al seu home. Això farà possible que apache es pugui identificar com a root a localhost mitjançant ssh.

Tot aquest procediment s'ha intentat fer el més transparent possible, encara que durant la instal·lació l'usuari haurà d'executar alguna comanda paral·lelament, tal com es demana a l'script.

#### 2.5.4 Aspectes de seguretat

Per a reforçar la seguretat del sistema una vegada instal·lat Drupal, s'han de tenir en compte els aspectes que es comenten en els següents punts. Aquí es tractaran aspectes de Drupal com modificar la contrasenya per defecte dels usuaris, i aspectes d'algun component necessari, com ara la contrasenya de l'administrador del servidor de bases de dades.

- Una vegada finalitzada la instal·lació de Drupal, ja es pot accedir a la web a través de l'adreça *http://localhost/drupal* des del mateix servidor o a través de *http://172.16.104.44/drupal* des d'un altre ordinador de la xarxa. A l'entorn web queden registrats 3 usuaris administradors: admin,

cluster i contacte amb el mateix login i contrasenya; el més recomanable és que abans de tot cadascun d'ells reforci aquestes contrasenyes.

- A l'instal·lador de Drupal s'ha cregut oportú forçar a l'administrador a crear una contrasenya per a l'usuari root de mysql, en cas de no tenir-ne cap d'assignada. És una forma d'assegurar-se que l'accés a les bases de dades queda protegit en el cas de no estar-ne fins el moment.
- Per a poder afegir la màquina local a l'arxiu de hosts coneguts i crear les claus de l'usuari apache, durant la instal·lació s'habilita el shell per a l'usuari apache. Com que l'usuari apache es considera un usuari del sistema, per defecte té inhabilitat logejar-se al sistema i, per tant, no es podrien crear les seves claus. Una vegada creades les claus se li torna a inhabilitar la shell a aquest usuari, i ja no podrà ingressar al sistema, ni tan sols amb la clau que se li hagi donat durant la instal·lació. Aquest usuari només tindrà l'accés per ssh a localhost.
- L'arxiu *settings.php* guarda la configuració d'alguns aspectes de Drupal, entre ells l'usuari i contrasenya que permeten accedir a la base de dades que s'utilitza. Aquest arxiu té permissos de lectura per a tots els usuaris del sistema (-rwxr-r-), el que permetria a qualsevol usuari donat d'alta al sistema operatiu, obtenir la contrasenya del propietari de la base de dades i accedir-hi sense restriccions. Com que aquest arxiu només és necessari que sigui accessible per l'usuari que llença Apache, s'han canviat els permissos a -rwx— (chmod 700), de forma que només hi tingui accés, a més de l'administrador del sistema del cluster, l'usuari propietari de l'arxiu. Degut a això, durant l'script d'instal·lació de Drupal també es modifica el propietari de l'arxiu, amb la comanda *chown*.
- Des l'entorn web es permet enllaçar al cluster de docència *AEC2* per a monitoritzar els seus recursos a través del Web Frontend Ganglia que té instal·lat. Per a poder connectar-se al cluster, el seu servidor haurà d'acceptar connexions TCP al port 80, i així oferir el servei web. Al tractar-se d'un servidor pot ser que tingui alguna regla de tallafocs que rebutji aquest tipus de connexions. Aquest problema es pot solucionar aplicant la següent regla d'*Iptables* en el servidor d'aquest cluster (172.16.3.100):

```
iptables -A INPUT -p tcp -m tcp -dport 80 -j ACCEPT
```

## 3 Desenvolupament

En aquest capítol s'explica quines solucions s'han definit a l'hora de desenvolupar el projecte. S'explica què són els mòduls dins de Drupal [23] i com s'han utilitzat, tant per a afegir funcionalitat com seguretat a l'entorn, a més de la forma en què s'integra l'entorn amb l'eina *Ganglia*.

### 3.1 Mòduls

Els mòduls ofereixen un ampli conjunt de funcions que permeten dotar l'entorn d'unes característiques que potser no queden cobertes en el primer moment que s'instal·la Drupal. Aquests mòduls addicionals permeten afegir noves funcions sense haver de modificar els mòduls del nucli de Drupal. Tot i això, en el cas d'haver de fer petites alteracions també es pot optar per modificar els mòduls del nucli. Una altra opció és descarregar un mòdul addicional i modificar-lo per adaptar-lo a les necessitats de l'entorn.

En aquest capítol es comentaran els mòduls que s'han modificat i els que s'han creat des de zero, detallant de quines funcions s'encarrega cadascun d'ells.

La àmplia documentació disponible a la web de Drupal ha permès un bon aprenentatge en la creació de mòduls, de la que se'n fa una introducció en l'Annex B, per ajudar a futures ampliacions d'aquest projecte. A més, en la documentació de la web es pot trobar tota la informació sobre la API de Drupal [31].

#### 3.1.1 Mòduls modificats

En aquest apartat es comenta quins han estat els mòduls modificats per afegir funcionalitats d'administració que Drupal no conté. Per a tal fi, s'han modificat mòduls que formen part del nucli (user) i d'altres addicionals (event, autologout).

##### **Mòdul User**

Habilita el sistema d'entrada i registre d'usuaris.

##### *Què fa?*

- Permet afegir, i modificar usuaris de l'entorn web.
- Permet afegir, i modificar rols d'usuari.
- Permet configurar els permissos i els paràmetres dels usuaris.

- Permet als usuaris canviar aspectes del seu compte personal (contrasenya, idioma, imatge, etc.).

#### *Què més hauria de fer?*

- Permetre afegir i usuaris al sistema que controla al cluster.
- Desactivar l'opció "Demanar una contrasenya nova" de l'entrada d'usuaris.
- Permetre assignar un TFC i director de TFC al moment que es crea un usuari.
- Permetre veure el títol i el nom del director del TFC al accedir al perfil dels usuaris.
- Restringir la opció de modificar la imatge personal. Només hauria de permetre aquesta opció a determinats usuaris.

#### *Què s'ha canviat?*

- S'ha modificat el seu fitxer principal, l' *user.module*, afegint un checkbox que permet l'opció d'afegir un usuari al cluster en el moment que es dóna d'alta a l'entorn web. Aquest checkbox també es mostra a l'hora d' un usuari. Quan es selecciona el checkbox es fa una crida a les comandes del sistema *useradd* o *userdel* per dur a terme aquesta operació.
- Com que a través de l'entorn web no s'ofereix la possibilitat de fer desaparèixer l'opció de demanar una contrasenya nova pels usuaris, s'han comentat algunes línies de codi del mateix fitxer.
- S'han afegit dos camps de text per assignar TFC i director de TFC pels usuaris que es donguin d'alta. Al començar a escriure al camp *director de TFC* apareixeran tots els professors que comencin per aquella lletra.
- Quan un alumne tingui assignat un TFC i/o un director, si s'accedeix al seu perfil apareixerà el nom del títol del TFC i el nom del director assignat.
- S'ha afegit el permís *Canviar la imatge d'usuari*. Des de la pàgina de Control d'accés es pot seleccionar a quins rols d'usuari se'ls permetrà accedir a aquesta opció. S'ha cregut oportú donar accés només a professors i administradors.

**Mòdul Event**

Habilita la creació d'esdeveniments amb visualització d'un calendari.

***Què fa?***

- Permet la creació de contingut mitjançant esdeveniments.
- Incorpora la possibilitat de mostrar un calendari remarcant les dates on es produeixen els esdeveniments.
- Incorpora un bloc on es mostra un resum dels esdeveniments que són propers a la data actual.

***Què més hauria de fer?***

- Permetre la creació de tasques individuals i tasques de grup.
- Mostrar les tasques individuals i tasques de grup al calendari d'esdeveniments.
- Mostrar les tasques individuals i tasques de grup al bloc de *Pròxims esdeveniments*.

***Què s'ha canviat?***

- S'han afegit el camp *rid* a la taula event de la base de dades que juntament amb l'*uid* (identificador d'usuari) permet diferenciar quan el contingut és de tipus esdeveniment, tasca o tasca de grup.
- Quan uid=0, rid=0 es tracta d'un esdeveniment.
- Quan uid=uid, rid=0 es tracta d'una tasca d'usuari.
- Quan uid=0, rid=rid es tracta d'una tasca de grup.
- En el moment d'emmagatzemar o mostrar el contingut, es tenen en compte aquests valors per comprovar si els usuaris poden realitzar l'operació.

### **Mòdul Autologout**

Afegeix temps de fi de sessió.

#### *Què fa?*

- Permet configurar la durada màxima de les sessions dels usuaris.
- Quan es detecta aquest temps d'inactivitat d'un usuari, la sessió queda finalitzada i s'envia a la pàgina d'entrada d'usuaris.
- Està implementat per que aquest temps no afecti a l'usuari administrador.

#### *Què més hauria de fer?*

- Permetre el fi de sessió automàtic per a l'administrador.

#### *Què s'ha canviat?*

- S'ha modificat l'arxiu *autologout.module*, permetent ara que el temps màxim d'inactivitat afecti també a l'administrador.

### **3.1.2 Mòduls creats**

En aquest apartat es detallen els mòduls que s'han creat per a poder dotar a l'entorn d'unes determinades característiques. Com que la majoria de mòduls existents per a Drupal estàn enfocats a funcions pròpies de l'entorn web, interactuen amb la seva base de dades i no incorporen eines d'administració del sistema operatiu. Arribats a aquest punt s'han creat aquests nous mòduls que permeten operar amb el sistema operatiu per realitzar funcions com crear còpies de seguretat o afegir usuaris al sistema.

### **Mòdul Contacte**

Permet contactar entre usuaris i administrador.

#### *Què ha de fer?*

- Crear un canal de comunicació entre els usuaris de l'entorn i l'administrador.
- Permetre als usuaris enviar missatges a través d'un formulari.
- Permetre als usuaris rebre missatges de l'administrador.

- Permetre a l'administrador de missatgeria enviar, rebre i missatges .
- Oferir la possibilitat de notificar la rebuda d'un missatge nou en totes les pàgines de l'entorn que es visitin.
- Oferir la possibilitat de notificar a través de correu electrònic la rebuda de missatges nous.
- Permetre configurar el nombre de missatges vistos per pàgina.
- Permetre visualitzar l'estat dels missatges enviats, és a dir, si han estat llegits pel receptor.
- Permetre visualitzar els missatges com una conversació, quan es tracti d'una resposta a un missatge enviat.

#### *Com s'ha implementat?*

- Per a gestionar el servei de missatgeria s'ha creat el rol d'usuari contacteadmin. Així existirà un administrador (usuari anomenat contacte) amb aquest rol, dedicat exclusivament a intercanviar missatges amb la resta dels usuaris.
- Només s'ha donat l'opció d' missatges a l'administrador de missatgeria.
- S'ha afegit l'opció *Contacte* al menú de navegació, que enllaça directament al llistat de missatges de l'usuari, i permet la redacció de missatges. Si es reben missatges nous hi han missatges d'entrada que encara no s'han llegit, al costat de la paraula *Contacte* apareixerà entre parèntesi el nombre d'aquests missatges.
- Quan un missatge s'envia com a resposta d'un altre missatge, aquests missatges queden vinculats. Quan es visualitza un missatge d'aquest tipus apareix l'opció *Veure conversació* sota del llistat de missatges. Això permet accedir a l'historial d'aquest "tema de conversació".
- Per a la configuració dels aspectes del mòdul (visualització de l'estat del missatge i nombre de missatges per pàgina) s'ha creat l'opció *Contacte* al submenú Configuració del web, dins del menú d'Administració. Aquest menú només és accessible per l'administrador de l'entorn, per tant el l'administrador de missatgeria no pot modificar aquests paràmetres.

- S'han creat les opcions *Notificació agressiva de missatges nous* i *Notificació e-mail de missatges nous* dins del menú Compte d'usuari. D'aquesta forma cada usuari podrà activar aquestes opcions segons li convingui. Per a poder rebre el correu electrònic de notificació de missatges nous, l'usuari ha d'haver inserit la seva adreça de correu electrònic.
- S'han creat els arxius d'idiomes per a català, castellà i anglès.

**Mòdul Cluster:**

Afegeix tasques pròpies d'un cluster de computació distribuïda.

***Què ha de fer?***

- Permetre la realització de còpies de seguretat del sistema.
- Permetre la programació de les còpies per a un altre moment.
- Permetre la visualització de les còpies programades pendents de realitzar.
- Permetre creació de grups al sistema operatiu.
- Permetre l'nació de grups del sistema operatiu.
- Permetre afegir usuaris als grups del sistema operatiu.
- Permetre usuaris dels grups del sistema operatiu.
- Mostrar l'últim accés dels usuaris donats d'alta al cluster.
- Mostrar l'espai de disc que ocupen els usuaris donats d'alta al cluster.
- Mostrar el grup al que pertanyen els usuaris donats d'alta al cluster.
- Donar accés als professors per veure tots els TFC que tenen assignats.

***Com s'ha implementat?***

- S'ha creat el menú *Backup* a la barra de navegació esquerra, que enllaça amb el formulari per fer les còpies de seguretat. Aquest menú només serà accessible per als usuaris amb el permís *crear còpies de seguretat* activat.
- En aquest formulari s'han de seleccionar la carpeta que es vol copiar i la carpeta on s'emmagatzemarà aquesta còpia. La còpia quedarà en un arxiu comprimit (*.tar.gz*), del que també es pot escriure el nom.



- En cas de programar la còpia, s'han d'especificar tant la data (dia/mes/any) com l'hora en que es realitzarà.
- S'ha creat el submenú Pendants, on es mostrarà les còpies programades que encara s'hagin d'efectuar. Per cada còpia programada es mostrarà l'usuari que l'hagi programada, la data, l'hora, i els arxius d'origen i destí.
- Cada vegada que es faci una còpia programada, aquesta queda automàticament eliminada de la base de dades, així s'evita ocupar espai innecessàriament.
- Per a afegir un usuari a un grup del sistema operatiu, s'ha creat un formulari accessible des de la pàgina d'edició del perfil d'aquest usuari. Aquest formulari només és accessible per als usuaris administradors, i també permet la creació de nous grups del sistema i l'eliminació de grups, a través d'una llista que mostra els grups ja existents.
- Per a mostrar la informació del sistema dels usuaris (últim accés, espai de disc ocupat i grup) s'ha creat l'apartat anomenat *Control del Sistema*. Aquest apartat es mostra al perfil de cada usuari, i és accessible pels usuaris que tinguin activats els permissos *accés a control del sistema* i *accedeix als perfils d'usuari*.
- S'ha creat el menú TFC a la barra de navegació que enllaça amb una pàgina on es mostren quins TFC té assignats l'usuari (professor). En el llistat es mostra l'alumne i el títol del TFC, i des d'aquí es pot accedir al perfil de l'alumne fent click sobre el seu nom. Per a accedir a TFC s'ha creat el permís *accés a TFC*, i s'ha assignat al rol *professors*.
- S'han creat els arxius d'idiomes per a català, castellà i anglès.

### 3.1.3 Tasques d'administració

Per a determinades tasques s'ha d'accedir al sistema i executar comandes que normalment utilitza l'administrador del sistema. Aquestes tasques es mostren en la Taula 3 .

Tasca	Comanda	Mòdul	Tipus mòdul
usuari del sistema	userdel	user	nucli
Afegir usuari al sistema	useradd	user	nucli
Crear grup al sistema	groupadd	cluster	addicional
grup del sistema	groupdel	cluster	addicional
Obtenir grup d'un usuari	cat /etc/group	cluster	addicional
Modificar grup d'un usuari	usermod	cluster	addicional
Obtenir espai de disc d'un usuari	du	cluster	addicional
Obtenir últim accés d'un usuari	lastlog	cluster	addicional
Programar còpia de seguretat	at	cluster	addicional

Taula 3: Tasques d'administració

Per a poder executar aquestes comandes des del servidor web es fa ús del protocol ssh. Cada vegada que s'hagi d'executar qualsevol d'aquestes comandes, l'usuari que ha llençat el servidor web (normalment apache) s'identifica al sistema operatiu com a l'usuari administrador *root*, mitjançant ssh, i executa la comanda. En la majoria dels casos les funcions que s'utilitzen es criden des del mateix codi php de la pàgina web, sense la necessitat d'haver de fer crides a scripts externs. El cas de la comanda *useradd* és un altre, degut a algunes circumstàncies que s'expliquen en el capítol 3.2.1.

## 3.2 Gestió de la seguretat

Degut a les característiques de l'entorn que s'ha creat, s'ha de tenir especial cura en la seguretat. Per a les tasques descrites en l'apartat 3.1.3, es crea un vincle entre l'entorn web i el sistema que controla el cluster, en el que es tenen privilegis d'administrador. Si un usuari malintencionat trobés una bretxa de seguretat en aquest punt faria que la seguretat de tot el cluster quedés compromesa. Per a minimitzar aquest risc, a més dels aspectes descrits a l'apartat 2.5.4, s'han implementat una sèrie de mesures de seguretat, com ara evitar la injecció de codi o evitar que qualsevol usuari s'oblidi una sessió oberta.

### 3.2.1 Contrasenyes d'usuaris

En aquest apartat no es comenta una mesura preventiva, sino un problema que ha aparegut en el moment d'assignar les contrasenyes als usuaris que es creen al sistema que controla al cluster, i per tant afecta a la seva seguretat.

Linux utilitza el sistema d'encryptació md5 per a gestionar les contrasenyes dels

usuaris, emmagatzemades a l'arxiu */etc/shadow*. Aquest sistema utilitza una funció hash irreversible que retorna una cadena de 128 bits a partir d'una cadena de text pla. Per a realitzar aquesta funció hash s'utilitza una llavor, a partir de la qual es realitzaran una sèrie d'operacions durant el procés de conversió. El resultat final, emmagatzemat per a cada usuari a l'arxiu esmentat és el següent:

```
$1$xxxxxx$yyyyyyyyyyyyyyyyyyyy
```

on les *x*'s correponen a la llavor utilitzada i les *y*'s a la sortida de la funció hash. Un exemple real utilitzant la llavor \$HTj4ofwK és:

```
$1$HTj4ofwK$vNQ59IdqagnywD5YAxBYV0
```

Inicialment, dins del mòdul User es va implementar un codi que permetia afegir usuaris al sistema del cluster a l'hora que eren creats a l'entorn web. Aquesta línia de codi és:

```
system("ssh root@localhost useradd $usuari -m -p $contrasenya");
```

D'aquesta forma s'assignava a l'usuari la mateixa contrasenya per a l'entorn web i al sistema del cluster.

El problema que va aparèixer és que amb la contrasenya no es permetia l'entrada al sistema del cluster, mentre que per l'entorn web sí que funcionava. Es va detectar que el motiu de l'error era degut als símbols \$, que serveixen per a refererir-se a les variables a PHP. Al executar la comanda *useradd* mitajçant la funció *system*, *ssh* no sabia interpretar els caràcters \$ dins la cadena i li assignava a l'usuari una contrasenya errònia.

Per a solucionar aquest problema s'ha creat l'script *useradd.php*, que es crida des del mòdul User de la següent manera:

```
system("ssh root@localhost php /var/www/html/drupal/scripts/useradd.php  
$name $pass");
```

En aquest cas en comptes de cridar la comanda *useradd* directament, s'utilitza aquest script PHP per a poder tractar les cadenes de text adequadament.

L'script *useradd.php* s'ha ubicat a la carpeta d'scripts de Drupal. Aquesta carpeta necessita els permissos de lectura i execució per qualsevol tipus d'usuari. Això no suposa cap risc, ja que aquest script només crida a la comanda *useradd* de Linux, i aquesta només es pot executar en el cas de ser administrador del sistema. A continuació es mostra el codi de l'script:

```
#!/usr/bin/php
<?php
    $usuari=$argv[1];
    $pass=$argv[2];
    $hash=crypt($pass);
    $st1=substr($hash,3,8);
    $st2=substr($hash,12,22);
    system("useradd $usuari -m -p '$1$'$st1'$st2");
?>
```

En aquest script es passen com a paràmetres el nom d'usuari i la contrasenya que se li assigna. Aquesta contrasenya es passa a la funció `crypt` per a encriptar-la de forma que Linux la pugui reconèixer. Per a separar els caràcters `$` de la resta de la cadena, s'utilitza la funció `substr`, seleccionant els caràcters segons posició que ocupen. Finalment, en el moment de fer la crida a `useradd` s'afegeixen els caràcters `$` individualment, de forma que es puguin interpretar correctament.

### 3.2.2 Dades de formularis

Cada vegada que s'utilitzen formularis, s'ha de tenir en compte que un usuari pot entrar dades al sistema. Si es tracta d'un usuari malintencionat pot estar intentant entrar dades que el sistema no espera rebre i crear una situació no esperada. La injecció de codi tracta precisament d'això, i podria permetre a un usuari qualsevol arribar a tenir accés al servidor com si es tractés de l'usuari administrador. Per exemple, si un usuari introdueix dins el camp els següents caràcters:

```
;rm -rf /home
```

L'interpret PHP quan llegeix el punt i coma, pot donar com a finalitzada la comanda actual i entendre que ha d'executar el següent codi que troba. Aquestes comandes s'estarien executant amb l'usuari que llença *Apache*, que segons el sistema podria tenir més o menys privilegis, segons els permissos que se li donguin i el grup del que formi part.

Per evitar això, en tots els formularis s'ha de comprovar que les dades entrades tenen un format correcte, i evitar l'entrada d'alguns caràcters que podrien arribar a comprometre el sistema.

Al formulari que permet fer còpies de seguretat del mòdul *cluster*, s'han creat camps de text per permetre introduir a l'usuari les carpetes origen i destí, i

l'arxiu destí de la còpia. Per a comprovar que les dades introduïdes en aquests camps són correctes, s'han utilitzat dues funcions que incorpora el llenguatge *PHP*, *file\_exists* i *is\_dir* quedant el següent codi:

```
if (!file_exists($form_values['source']) && !is_dir($form_values['source']))
{
    form_set_error('source', t('Source: You must enter an existent folder.'));
}
```

Amb aquesta sentència es comprova que les dades introduïdes al camp del formulari *source* són correctes, tractant-se d'un arxiu que existeix i que a més, es tracta d'un directori.

En el moment d'especificar la data i hora de la còpia, només és necessari comprovar que s'estan intrduïnt valors numèrics, i que siguin correctes, com en el cas del camp de les hores:

```
if (!is_numeric($form_values['hours']) || $form_values['hours'] < 0 ||
    || $form_values['hours'] > 23)
{
    form_set_error('specify', t('You must enter a valid time.'));
}
```

Un últim exemple d'aquest mateix mòdul, és el formulari que permet la gestió de grups del sistema. Aquí es troba un camp de text on es pot introduir el nom del grup que es vol crear. A part de comprovar si el nom del grup ja existeix, es comprova que els caràcters introduïts siguin vàlids per a un nom de grup. Això es logra creant la funció *check\_group*:

```
function check_group($groupname)
{
    if (ereg("[a-zA-Z0-9\-\_]{1,25}$", $groupname)) { return TRUE; }
    else { return FALSE; } // Nom incorrecte
}
```

En aquest cas s'utilitza la funció de *PHP* *ereg*, que permet comprovar tant els tipus de caràcters, com la longitud del nom del grup.

### 3.2.3 Cookies

Una cookie (galeta) és un fragment d'informació que s'emmagatzema al disc dur del visitant d'una pàgina web a través del seu navegador, a petició del servidor de la pàgina. Aquesta informació pot ser recuperada pel servidor en properes visites. Els usos més freqüents de les cookies són:

- Portar el control d'usuaris: Quan un usuari introdueix el seu nom d'usuari i contrasenya, s'emmagatzema una cookie amb aquestes dades per evitar haver d'introduir-les cada vegada que es connecti al servidor.
- Configurar opcions de disseny (colors, fons, etc.) o de continguts al visitant.
- Obtenir informació sobre hàbits de navegació de l'usuari (spyware).

S'ha de tenir en compte que una cookie no identifica a una persona, sino a una combinació de navegador i ordinador. Això pot portar problemes en sistemes monousuari on el navegador estigui configurat per acceptar totes les galetes. En aquest cas, si un usuari tanca el navegador sense haver finalitzat la sessió, quedarà emmagatzemada la galeta amb l'identificador d'aquest usuari. En el moment que un altre usuari entri al sistema i accedeixi amb el navegador a la mateixa pàgina web, estarà identificat com a l'usuari anterior que no ha finalitzat la sessió.

Els navegadors també tenen un paper important, perquè igual que permeten denegar l'acceptació de galetes, també permeten recordar i obrir automàticament les últimes pàgines web visitades. Així doncs, no es pot córrer el risc de deixar la seguretat en mans de la configuració dels navegadors web dels usuaris.

Per resoldre aquest problema en el cas de l'entorn web implementat, s'utilitza la combinació de 2 solucions:

1. *settings.php*: Arxiu ubicat a la carpeta */var/www/html/drupal/sites/default* del servidor web, des del qual es poden configurar alguns paràmetres del servidor. Inicialitzant la variable *session.cookie\_lifetime* a zero, es força a donar validesa a les cookies actuals només fins que el navegador es tanqui.
2. Mòdul Autologout: Aquest mòdul permet configurar la durada màxima de les sessions dels usuaris una vegada es detecta inactivitat. S'ha configurat per donar un temps màxim d'inactivitat de 180 segons, després d'aquest temps es finalitza la sessió de l'usuari. Aquest temps es pot modificar de la pàgina de configuració del mòdul.

Amb la combinació d'aquestes 2 solucions queda resolt aquest problema de seguretat en sistemes monousuari.

En el cas de sistemes multiusuari aquests problemes de seguretat no apareixen, ja que la informació de les galetes queda emmagatzemada en l'espai de disc de l'usuari, al que només i té accés el propi usuari.

### 3.3 Integració amb Ganglia

El *Ganglia Web Frontend* permet la monitorització de clusters a través de pàgines web dinàmiques. Per a integrar aquest servei a l'entorn web implementat, es fa ús d'una eina d'administració que ofereix Drupal, els blocs.

Els blocs són caixes de contingut que es poden mostrar en certes regions de les pàgines web, per exemple, en els costats. Normalment es generen automàticament pels mòduls, però els administradors també poden crear-ne manualment des de la pàgina d'administració dels blocs, tal com es mostra en la Figura 10.

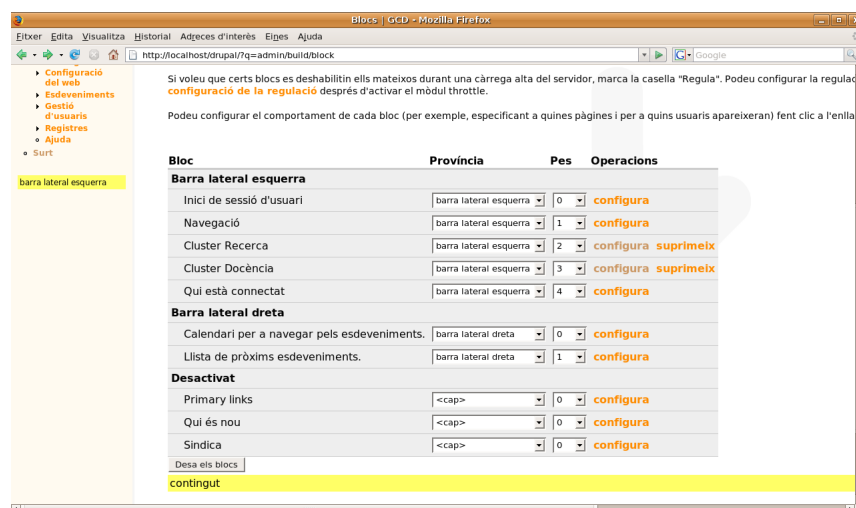


Figura 10: Administració dels blocs

Només es mostren els blocs actius, i es poden col·locar especificant a quina àrea de la pàgina haurien d'aparèixer. Les etiquetes il·luminades en aquesta pàgina mostren les regions en les que els blocs es poden mostrar. Es pot especificar on apareixerà un bloc dins una regió ajustant el seu pes.

Es pot configurar el comportament de cada bloc (per exemple, especificant a quines pàgines i per a quins usuaris apareixeran) fent click a l'enllaç "configura" per a cada bloc. A la Figura 11 figura es pot veure un exemple de configuració d'un bloc:



Figura 11: Configuració dels blocs

En aquest cas es tracta del bloc anomenat cluster Recerca. Al cos del bloc s'ha d'inserir el contingut del bloc que es mostrarà als usuaris, en llenguatge HTML o PHP. Per tal de mostrar un enllaç a la pàgina del frontend web d'aquest cluster es crea una referència a aquesta pàgina, i s'omplen els espais títol i descripció del bloc. Abans de desar els canvis s'especifica per a quins rols d'usuari es mostrarà el bloc.

Una vegada configurat el bloc per a cada cluster, s'especifica en quina regió apareixerà cadascun, i quin pes se'ls hi dóna. Els blocs resultants d'aquestes operacions es poden observar renarcats amb un rectangle vermell, en la barra lateral esquerra de la Figura 12 .



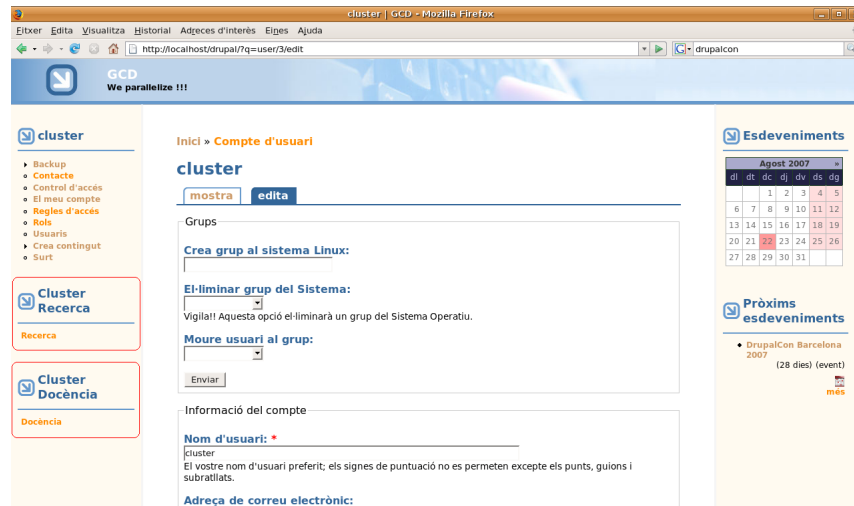


Figura 12: Resultat dels blocs

D'aquesta forma, en pocs passos, s'ha resolt la integració del *Ganglia* amb l'entorn web. Gràcies a l'utilització dels blocs aquesta feina ha resultat molt senzilla, havent d'afegir només una línia de codi HTML i configurar algun aspecte del bloc.

## 4 Conclusions i treball futur

El treball ha complert els objectius previstos, ja que s'ha ajustat a tot el que es demanava des d'un inici. S'ha creat un entorn web capaç de realitzar tasques d'administració d'un cluster i monitoritzar l'estat dels seus recursos en temps real.

Gràcies a aquest treball s'han ampliat els coneixements adquirits en les àrees d'Internet (programació PHP, entorn web, coneixements client-servidor), Sistemes Operatius (scripting, seguretat, administració del sistema) i BBDD. A més s'han adquirit coneixements sobre el funcionament dels clusters de computació distribuïda.

Com a treball futur es poden realitzar les següents ampliacions:

- Ampliar la funcionalitat a l'administració de múltiples clusters.  
En l'entorn creat les tasques referents a l'administració del cluster s'efectúen sobre un únic cluster. Modificant el mòdul Cluster es pot afegir algun menú des del que seleccionar sobre quin cluster es volen aplicar les accions. El canvi a realitzar en aquest mòdul consisteix en modificar el codi de les crides al sistema del cluster i, en comptes d'actuar sobre la màquina *localhost*, fer-ho sobre la màquina corresponent.
- Incloure l'administració d'un gestor de cues mitjançant una interfície gràfica, que permeti realitzar feines com enviar tasques a la cua, eliminar tasques de la cua i mostrar l'estat de les tasques. Per a tal fi, es pot ampliar el mòdul Custer fent ús de l'eina *PSB*, mencionada a l'apartat 1.4.
- Afegir al mòdul Cluster una funcionalitat per a visualitzar els registres i logs del cluster, i així tenir més control sobre el què passa al cluster. Aquesta informació pot ser referent a registres i incidències dels nodes o informes d'usuaris, i es pot guardar en fitxers de text per després llegir-ne el contingut des de la web. També es poden crear estadístiques que ajudin a interpretar tota aquesta informació.

## Referències

- [1] <http://www.vergenet.net/linux/fake/> - Web del projecte Fake
- [2] <http://oscar.openclustergroup.org/> - Web del toolkit OSCAR
- [3] <http://www.rocksclusters.org/> - Web del toolkit Rocks
- [4] <http://logcheck.org/> - Web de l'eina LogCheck
- [5] <http://sourceforge.net/projects/logsentry> - Web del projecte LogSentry
- [6] <http://caspian.dotconf.net/menu/Software/LogDog> - Web de LogDog
- [7] <http://www.kernel.org/software/mon/> - Web de l'eina MON
- [8] <http://bb4.com> - Web de l'eina Big Brother
- [9] <http://clumon.ncsa.uiuc.edu> - Web de l'eina CluMon
- [10] <http://www.nagios.org> - Web de l'eina Nagios
- [11] <http://supermon.sourceforge.net/> - Web del projecte Supermon
- [12] <http://www.vergenet.net/linux/ldirectord/> - Web de l'eina ldirectord
- [13] <http://ultramonkey.org/> - Web de l'eina Ultra Monkey
- [14] <http://www.csm.ornl.gov/torc/C3/> - Web del toolkit C3
- [15] <http://openbps.org/> - Web de l'eina Portable Batch System
- [16] <http://www.cmsmatrix.org/> - Comparativa entre la majoria de CMS
- [17] <http://www.opensourcecms.com/> - Exemples de la majoria de CMS
- [18] <http://phpnuke.org/> - Web oficial de PHP Nuke
- [19] <http://postnuke.com/> - Web oficial de Postnuke
- [20] <http://www.mamboserver.com/> - Web oficial de Mambo
- [21] <http://www.joomla.org/> - Web oficial de Joomla!
- [22] <http://drupal.org/> - Web oficial de Drupal
- [23] <http://drupal.org/project/Modules/> - Mòduls de Drupal

- [24] <http://drupal.org/project/Themes> - Temes oficials de Drupal
- [25] <http://drupal.org/community> - Comunitats d'usuaris de Drupal
- [26] <http://drupal.cat/> - Grup d'usuaris de Drupal en llengua catalana
- [27] <http://www.drupal.org.es/> - Drupal Hispano
- [28] <http://www.onlamp.com/> - Web dedicada a la plataforma LAMP
- [29] <http://www.ganglia.info/> - Web del projecte Ganglia
- [30] <http://drupal.org/project> - Secció de descàrregues oficials de Drupal
- [31] <http://api.drupal.org/> - Informació de l' API de Drupal

## A Scripts d'instal·lació

### A.1 install\_gmetad

```
#!/bin/bash
if [ "$UID" -ne 0 ]
then
    echo "Error: Has d'executar la instal·lació com a root"
    exit
fi
# Eina rrdtool per a crear les imatges que mostra ganglia
echo "Instal·lant eina rrdtool (Connexió a internet necessària)"
yum -y install rrdtool rrdtool-devel gcc
mkdir -p /var/lib/ganglia/rrds
chown -R nobody /var/lib/ganglia/rrds
# Instal·lació ganglia
tar xvfz ganglia-3.0.4.tar.gz
cd ganglia-3.0.4/
./configure --with-gmetad
make
make install
# Arxiu configuració frontend -> gmetad.conf
cp ../gmetad.conf /etc
# Iniciarà dimonis al arrancar el servidor
cp gmond/gmond.init /etc/rc.d/init.d/gmond
/sbin/chkconfig --add gmond
/etc/rc.d/init.d/gmond start
cp gmetad/gmetad.init /etc/rc.d/init.d/gmetad
/sbin/chkconfig --add gmetad
/etc/rc.d/init.d/gmetad start
# Instal·lació web frontend
mkdir /var/www/html/ganglia
cp -r web/* /var/www/html/ganglia
# Esborrar residus
cd ..
rm -rf ganglia-3.0.4/
echo "Instal·lació finalitzada"
```

## A.2 install\_gmond

```
#!/bin/bash
# Script d'instal·lació gmond (nodes)
# Albert Ros Simon
if [ "$UID" -ne 0 ]
then
    echo "Error: Has d'executar la instal·lació com a root"
    exit
fi
tar xvfz ganglia-3.0.4.tar.gz
cd ganglia-3.0.4/
./configure --with-gmetad
make
make install
# Crear arxiu /etc/gmond.conf amb configuració per defecte
/usr/sbin/gmond -t > ./etc/gmond.conf
echo "Ara has de modificar els valors de les seccions host i cluster
de l'arxiu /etc/gmond.conf"
read
# Iniciarà el dimoni al arrancar el node
cp ./gmond/gmond.init /etc/rc.d/init.d/gmond
/sbin/chkconfig --add gmond
/etc/rc.d/init.d/gmond start
# Esborrar residus
cd ..
rm -rf ganglia-3.0.4/
echo "Instal·lació finalitzada"
```

### A.3 install\_drupal

```
#!/bin/bash
# Script d'instal·lació de Drupal
# Albert Ros Simon
if [ "$UID" -ne 0 ]
then
    echo "Error: Has d'executar la instal·lació com a root"
    exit
fi
# Dades de l'usuari que llença el servidor Apache
usuari="apache"
home_apache="var/www"
home_www="/var/www/html"
shell="/sbin/bash"
tar xvfz drupal.tar.gz -C $home_www
# Es canvia usuari propietari a arxiu de configuració
chown $usuari $home_www/drupal/sites/default/settings.php
clear
echo "Per seguretat s'ha de protegir el compte de root de mysql."
echo "Si root no té contrasenya, assigna-li una així:"
echo "Ordre: mysqladmin -u root password contrasenya_root"
echo "Introdueix la contrasenya de root de mysql:"
read passmysql
mysql --default-character-set=utf8 -u root -p$passmysql < drupaldb.sql
# Instal·lació openssh-server i connectors necessaris
yum -y install openssh-server php-mysql php-mbstring php-mcrypt
# Activar shell per a poder entrar com a usuari i crear claus
/usr/sbin/usermod --shell /bin/bash $usuari
# Crear carpeta per guardar claus
mkdir -p $home/.ssh
chown $usuari $home/.ssh
echo "Escriu una contrasenya per l'usuari $usuari"
passwd $usuari
clear
echo "Obre un altre terminal i escriu:"
echo "su $usuari (i escriu la contrasenya que li acabes de donar)"
```

```
echo "ssh localhost (per afegir localhost a know_hosts)"
echo "escriu 'yes'"
echo "introdueix contrasenya donada"
echo "ssh-keygen -t dsa (per a crear claus de $usuari)"
echo "enter (per a ometre directori on guardar les claus)"
echo "enter (per no utilitzar passphrase)"
echo "enter (per reintroduir passphrase buit)"
echo "exit (per sortir de ssh)"
echo "exit (logout)"
echo "Prem una tecla per finalitzar...."
read
# Afegir clau pública per entrar per ssh com a usuari root
cat $home_apache/.ssh/id_dsa.pub > > /root/.ssh/authorized_keys
# Deshabilitar el login de l'usuari que llença Apache
usermod -lock -shell /sbin/nologin $usuari
# Connector php-mysql necessari
clear
echo "Instal·lació finalitzada"
echo "Ja pots accedir a http://localhost/drupal des del navegador"
echo "Els usuaris/contrasenyes creats per defecte són:"
echo "admin/admin"
```



## B Ampliacions mitjançant mòduls

En el moment que es vulgui afegir alguna nova funcionalitat a la web, es podrà fer afegint un mòdul addicional. En aquest punt existeixen dues opcions, buscar-ne un ja existent o crear-ne un de nou.

En el primer cas, el més recomanable és acudir a la secció de mòduls de la web oficial de Drupal, on s'ofereix la possibilitat de descarregar gran quantitat de mòduls. Es troben ordenats per categories, per nom o bé per data i s'haurà de descarregar la versió per Drupal 5.x, ja que els de versions anteriors són incompatibles al tenir una altra estructura d'arxius.

Una vegada descarregat l'arxiu comprimit (.tar.gz) s'haurà de descomprimir a la carpeta `/var/www/html/drupal/sites/all/modules`, ja que la carpeta `/var/www/html/drupal/modules` es reserva per als mòduls del nucli.

Si no es troba un mòdul adequat, s'haurà d'optar per l'opció de crear-ne un de nou. Els passos a seguir són els següents:

1. Crear una carpeta amb el nom que se li vulgui donar al mòdul dins de `/var/www/html/drupal/sites/all/modules`
2. Crear un arxiu dins de la carpeta creada, amb el nom `'nom_mòdul.info'`. Aquest arxiu conté la informació bàsica que es mostra en el menú d'administració dels mòduls, com el nom, descripció, paquet o versió de mòdul.
3. Crear un arxiu amb el nom `'nom_mòdul.install'`. Aquest arxiu només és necessari si el mòdul que es vol crear necessita una nova base de dades. En aquest cas dins de la funció `'nom_mòdul_install()'` s'introduirà el codi en SQL per crear la amb els seus camps. És recomanable afegir la funció `'nom_mòdul_uninstall()'` per a esborrar la creada en cas de desinstal·lar el mòdul. Així no queden residus en la base de dades de Drupal.
4. Crear un arxiu amb el nom `'nom_mòdul.module'`. Dins aquest arxiu es declararan les funcions anomenades hook, aquestes funcions internes serveixen per controlar alguns aspectes del mòdul que es crearà. En el cas que no sigui necessari utilitzar les característiques d'alguna funció hook, no serà necessari incloure-la en el fitxer del mòdul. Alguns dels hooks més utilitzats són:

### ***hook\_perm()***

En aquest hook es declaren els diferents permissos que es podran atorgar

als usuaris sobre aquest mòdul. Així si es defineix:

```
function nom_mòdul_perm() { return array('llegir','escriure'); }
```

en qualsevol punt es podrà controlar si un usuari té permissos sobre un contingut fent la següent crida:

```
if (user_access('llegir') { ... }
```

Tot això implicarà que al menú d'administració dels mòduls, l' administrador tindrà la opció d' atorgar els permissos llegir i/o escriure a un rol d'usuaris.

### ***hook\_menu()***

Com el seu nom indica, dins d' aquesta funció es declararan tots els menús que es vulguin fer aparèixer en el mòdul. A continuació es mostra un exemple del mòdul cluster:

```
function cluster_menu() {
  $items = array ();
  $items[] = array (
    'path' => 'backup/pendents',
    'title' => t('Pendents'),
    'callback' => 'pendents_llista',
    'access' => user_access('crear backups'),
    'type' => MENU_NORMAL_ITEM,
  );
  return $items;
}
```

Aquest codi crearà un menú anomenat Pendents, que cridarà a la funció `pendents_llista` en el moment que es seleccioni la opció del menú. Aquest menú només es mostrarà pels rols d' usuari als que se'ls hagi atorgat el permís 'crear backups'.

### ***hook\_user()***

Aquí es pot afegir la informació que es vulgui mostrar o editar sobre un

usuari. Aquesta informació apareixerà quan qualsevol usuari accedeixi al menú 'El meu compte' o quan un usuari amb permissos per a gestió d'usuaris accedeixi al perfil de qualsevol usuari registrat.

### ***hook\_uninstall()***

Serveix per a fer una crida sql quan l'administrador decideix un mòdul del sistema, des del menú d'administració de mòduls. Una vegada l'elimina, es fa la crida a aquesta funció i s'executa el codi corresponent. Es una bona manera de mantenir neta la base de dades, ja que normalment el que es fa aquí és les taules que s'han creat a l'hora d'instal·lar el mòdul i que figuren en l'arxiu *nom\_mòdul.install*. A més, també serveix per variables de memòria, en el cas d'haver-ne utilitzat.

5. Una vegada creats aquests fitxers, el mòdul apareixerà en la pàgina d'administració de mòduls i, en alguns casos, serà necessari llençar l'script *update.php* des d'aquesta mateixa pàgina per a que el reconegui. En aquest moment ja es pot activar el mòdul per a aprofitar-ne les seves funcions.
6. El següent pas serà donar assignar permissos als usuaris que se'ls hi vulgui donar accés a determinades funcions del mòdul. Aquests permissos s'assignen segons els rols d'usuari, des de la secció Control d'accés del menú encarregat de la Gestió d'usuaris.
7. Com s'ha dit en el punt 4, molts mòduls afegeixen menús a l'entorn web per a accedir a noves funcionalitats. La posició on es vulgui mostrar aquest menú es configurarà en el menú d'administració de blocs. Drupal permet que aquests menús es puguin mostrar en diferents posicions de la pantalla: barra lateral esquerra, barra lateral dreta, contingut (part principal de l'entorn web), capçalera i peu. Normalment, la majoria de blocs es fan aparèixer en la barra lateral esquerra o dreta, per comoditat a l'hora de navegar per l'entorn.
8. En el moment que es vulgui desinstal·lar un mòdul, la opció més correcta de fer-ho és desmarcar el mòdul de la llista i desar la configuració. En aquest moment el mòdul estarà disponible en la llista de mòduls desinstal·lables, i ara sí que es podran totes les dades relacionades amb aquest mòdul. En aquest punt, si es desitja també es pot la carpeta d'aquest mòdul del disc, així ja no apareixerà més al llistat de mòduls, ni es donarà l'opció d'activar-lo de nou.